KPCI-811 多功能卡
使用说明书

北京科瑞兴业科技有限公司

北京科瑞兴业科技有限公司　地址：北京市海淀区知春里 28 号开源商务写字楼 212、213 室
邮政编码：100086　电话：010-51650651　010-62527214　传真：010-62657424
http://www.krxgk.com
Sales E-mail: sgq@krxgk.com　Tech Support E-mail: lilanzhen007@126.com

目录

第一章　概述

一、介绍

KPCI-811 卡是一种基于 PCI 总线的数据采集卡，可直接插在 IBM-PC/AT 或与之兼容的计算机内的任一 PCI 插槽中，可构成实验室、产品质量检测中心和大专院校等各种领域的数据采集、分析和数据处理系统。也可构成工业现场的过程监控系统。

◆ 32 位 PCI 总线，支持 PCI2.2 协议，真正实现即插即用
◆ FPGA 接口芯片设计，具有极高的保密性，特别适合 OEM 合作
◆ **100 K**Hz 12 位 A/D 转换器，通过率为 90K
◆ 16 通道单端模拟输入或 8 路双端模拟输入
◆ 支持软件查询方式、中断方式，两种方式的传输率均可达到 90K
◆ 1K 深度的 FIFO 保证数据的完整性
◆ 2 个独立的 16 位计数/定时器（82C54 芯片），信号与 TTL 电平兼容
◆ 2 个独立的 12 位数字/模拟量输出通道
◆ 16 路开关量输入通道输入电平与 TTL 兼容、16 路输出开关量输出通道电平与 TTL 兼容
◆ 完备的函数模块使您可以方便的编写自己的应用程序，而不需要对硬件有所了解，真正做到了即插即用

二、性能及技术指标

2.1 PCI 局部总线性能

▼PCI 总线宽度 32 位，同步工作频率可达到 33MHz，最高传输速率为 132MB/S
▼使用方便，能够实现自动配置，实现设备的即插即用
▼提供数据和地址奇偶校验功能，保证了数据的完整性和准确性

2.2 模拟信号输入部分

★ 模拟通道输入数: 16 路单端或 8 路双端输入
★ 模拟输入电压范围：±5V、0～＋5V、0～＋10V
★ 模拟输入阻抗:　10MΩ

2.3 A/D 转换电路部分

- A/D 分辨率：12Bit(4096)
- 非线性误差：±1LSB(最大)
- 转换时间：10us
- 系统测量精度：0.1%

2.4 A/D 采集方式及周期

■ 三种采样触发方式：定时触发、软件触发和外触发
■ 高精度采样周期、窄定时间隔（1/16 微秒），宽定时范围（1us-56us）

2.5 D/A 模拟量输出电路部分

- D/A 分辨率：12Bit(4096)
- 模拟输出电压、电流范围：±5V、±10V、0～＋10V 和 4~20mA 范围的电流输出
- 输出通道数：2 路
- 非线性误差：±1LSB(最大)
- 建立时间：10μS(0.01%精度)

- 输出阻抗：0.2Ω
- 具有加电清零功能

2.6 2 个独立的 16 位计数/定时器，信号与 TTL 电平兼容

- ♠ GATE 计数器的计数使能输入端内部已接有上拉电阻，以方便使用
- ♠ CLK、OUT 输出端满足标准 TTL 电气特性。

  低电平小于 0.4V，高电平大于 2.6V，OUT 的最大输出驱动电流 1 毫安。

2.7 开关量输入输出部分

- ◆ 16 路开关量输入通道输入电平与 TTL 兼容
- ◆ 16 路开关量输出通道输出电平与 TTL 兼容
- ◆ 输入 TTL 电平，吸入电流小于 0.5 毫安

2.8 FIFO 存储器

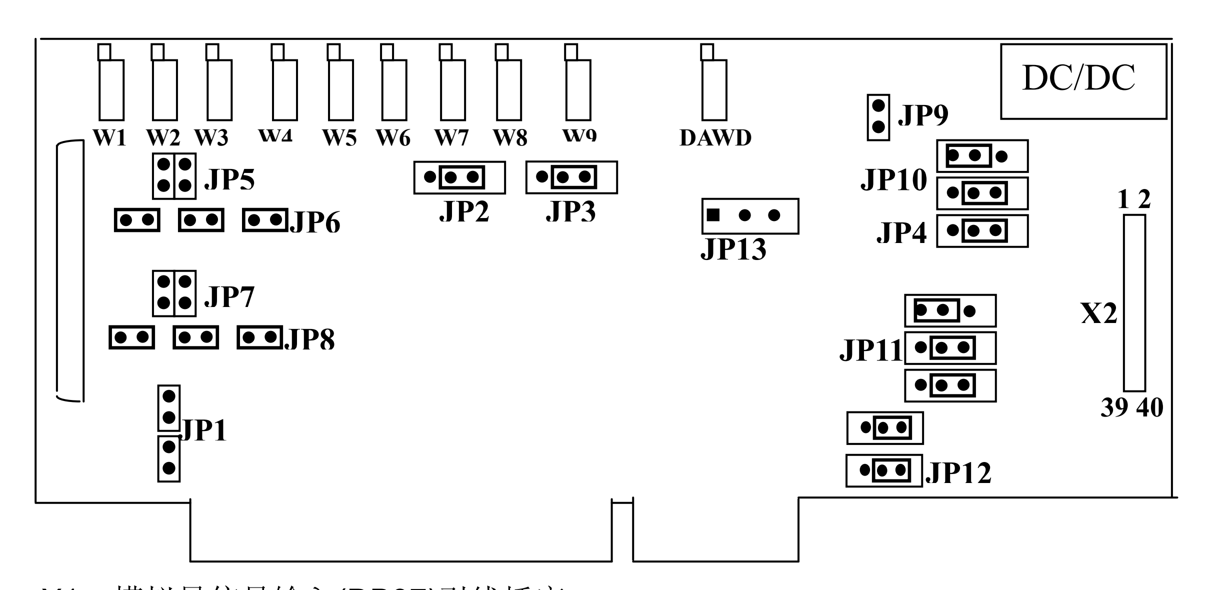
- 深度： 1K （可选配 1~ 32K words）
- 宽度： 12Bits
- 标志： 满、空、半满

三、软件支持

提供 Windows95/98/NT 下的多种语言的驱动，具有采集、显示、计数/定时、控制等功能；并提供在 VB 和 VC 环境下，开发的示例程序（详见软件说明部分）。

第二章 元件位置图、信号输出插座和开关跳线选择定义

一、 主要元件布局图



X1：模拟量信号输入(DB37)引线插座

X2：开关量输入/输出信号(40P 双排针)引线插座

W1：程控增益放大器零点调整电位器

W2：D/A 2 满度调整

W3：D/A 1 满度调整

W4：D/A 1 零点调整

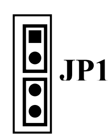
W5：D/A 2 零点调整
W6：运算放大器零点调整
W7：A/D 单极性零点调整
W8：A/D 满度调整
W9：A/D 双极性零点调整
W10：D/A 基准源调整

JP1：单/双端输入方式选择
JP2，JP3：模拟电压输入量程选择
JP5，JP6，JP9，JP10：D/A1 输出量程选择
JP7，JP8，JP9，JP11：D/A2 输出量程选择
JP12：计数器工作方式选择

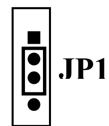
二、 短路套设置

2.1 模拟信号输入方式选择

① 单端输入方式：

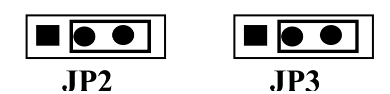


② 双端输入方式：

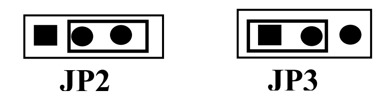


2.2 模拟信号输入量程选择

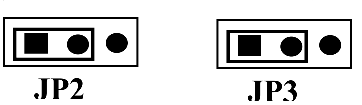
① 模拟量输入范围 0V～10V 的选择



② 模拟量输入范围为±5V 的选择

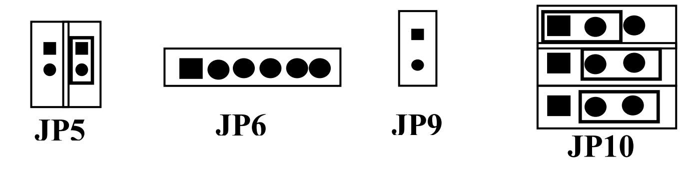


③ 模拟量输入范围为±10V 的选择图

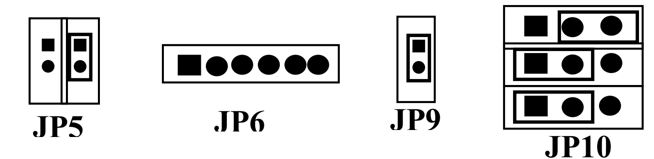


2.3 D/A 输出量程选择

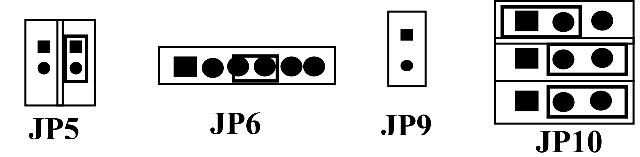
① 模拟量输出范围 **D/A 1** 0V～10V 的选择



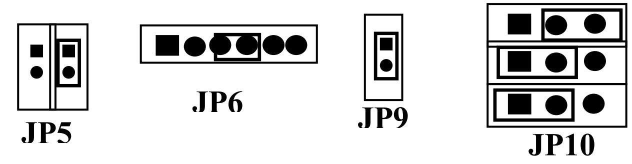
② 模拟量输出范围 **D/A 1** ±10V 的选择



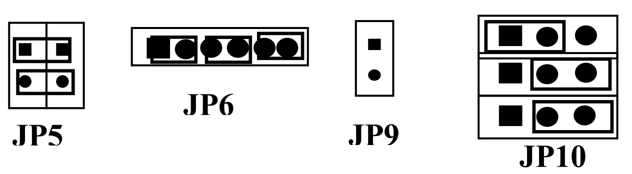
③ 模拟量输出范围 **D/A 1** 0V～5V 的选择



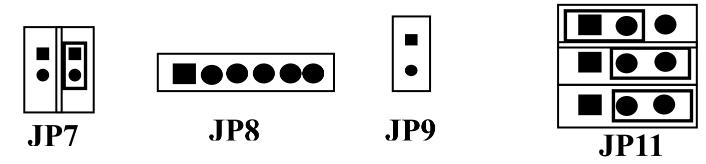
④ 模拟量输出范围 **D/A 1** ±5V 的选择



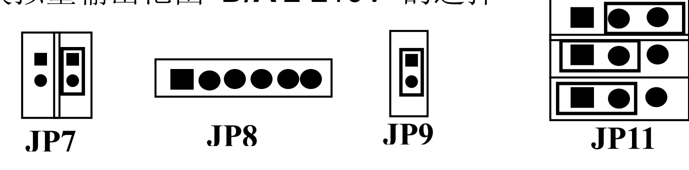
⑤ 模拟量输出范围 **D/A 1** 4～20mA 的选择



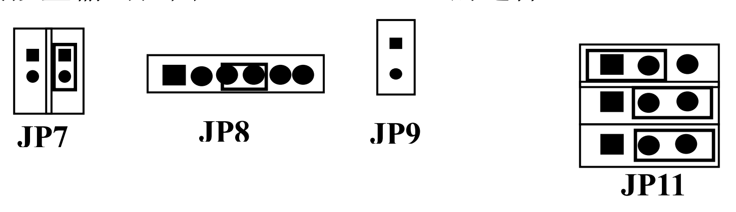
⑥ 模拟量输出范围 **D/A 2** 0V～10V 的选择



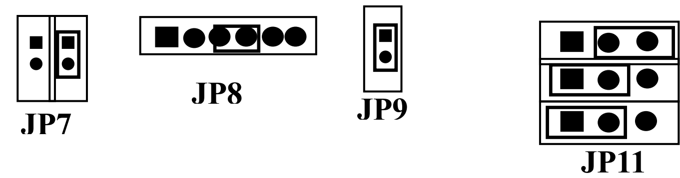
⑦ 模拟量输出范围 **D/A 2** ±10V 的选择



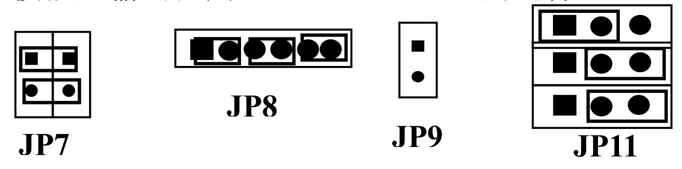
⑧ 模拟量输出范围 **D/A 2** 0V～5V 的选择



⑨ 模拟量输出范围 **D/A 2** ±5V 的选择

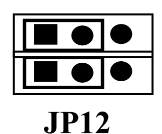


⑩ 模拟量输出范围 **D/A 2** 4～20mA 的选择

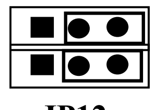


2.4 计数器工作方式选择

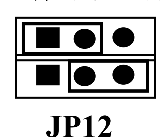
① 通道 0,1 工作于计数方式



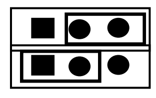
② 通道 0,1 工作于定时方式



③ 通道 0 工作于定时方式，通道 1 工作于计数方式



④通道 0 工作于计数方式，通道 1 工作于定时方式



三、信号输入输出插座定义

3.1 模拟量输入、输出接口信号定义（输入插座为DB37孔式弯针插座）

插座引脚号	引 脚 定 义	插座引脚号	引 脚 定 义
1	CH1 (In1+)	20	CH2 (In2+)
2	CH3 (In3+)	21	CH4 (In4+)
3	CH5 (In5+)	22	CH6 (In6+)
4	CH7 (In7+)	23	CH8 (In8+)
5	CH9 (In1-)	24	CH10 (In2-)
6	CH11 (In3-)	25	CH12 (In4-)
7	CH13 (In5-)	26	CH14 (In6-)
8	CH15 (In7-)	27	CH16 (In8-)
9	AGND	28	AGND
10	AGND	29	DA_V_Out1
11	AGND	30	DA_V_Out2
12	AGND	31	DA_I_Out1+
13	DA_I_Out1 -	32	DA_I_Out2+
14	DA_I_Out2 -	33	空闲
15	空闲	34	空闲
16	空闲	35	空闲
17	空闲	36	空闲
18	空闲	37	TR 外触发信号
19	外触发信号地		

3.2 计数器,开关量输入输出接口信号定义（插座为 40P 双排针插座）

插座引脚号	引 脚 定 义	插座引脚号	引 脚 定 义
1	Din1(开关量输入)	2	Din2(开关量输入)
3	Din3	4	Din4
5	Din5	6	Din6
7	Din7	8	Din8
9	Din9	10	Din10
11	Din11	12	Din12
13	Din13	14	Din14
15	Din15	16	Din16
17	Dout1(开关量输出)	18	Dout2(开关量输出)
19	Dout3	20	Dout4
21	Dout5	22	Dout6
23	Dout7	24	Dout8
25	Dout9	26	Dout10
27	Dout11	28	Dout12
29	Dout13	30	Dout14
31	Dout15	32	Dout16
33	CLK 1	34	CLK2
35	GATE1	36	GATE2
37	OUT1	38	OUT2
39	数字地	40	数字地

CH1～CH16： 单端模拟信号输入通道
In1+ ～In8+ ： 双端模拟信号输入正端
In1- ～ In8- ： 双端模拟信号输入负端
AGND：模拟地
DA_V_Out1,2： 模拟信号（电压）输出端
DA_I_Out1,2 +： 模拟信号（电流）输出正端
DA_I_Out1,2 -： 模拟信号（电流）输出负端
Din1 ～ Din16 ： 开关量输入通道
Dout1 ～ Dout16 ：开关量输出通道
CLK1,CLK2： 计数器输入端
GATE1,GATE2：计数器的计数使能输入端
OUT1,OUT2： 定时/计数器结束点输出
TR：外部触发信号，当TR有一符合TTL电平的由低至高的变化(上升沿)时，KPCI-811卡将按预先设定的采集通道总数进行采集，直至采集结束。程序举例见软件说明书相应部分。

3.3 当计数器、开关量输入输出接口使用 40P 扁平带缆插头转成 37 芯 D 型插头的转接带缆时，请注意：1、信号定义管脚的不同，2、计数器 2 的输出 OUT 没有了，3、开关量输入输出的数字地没有了，只能从 AD\DA 插座连接。转接带缆信号定义如下表：

计数器,开关量输入输出接口信号定义（插座为 37 芯 D 型插座）

插座引脚号	引 脚 定 义	插座引脚号	引 脚 定 义
1	Din1(开关量输入)	20	Din2(开关量输入)
2	Din3	21	Din4
3	Din5	22	Din6
4	Din7	23	Din8
5	Din9	24	Din10
6	Din11	25	Din12
7	Din13	26	Din14
8	Din15	27	Din16
9	Dout1(开关量输出)	28	Dout2
10	Dout3	29	Dout4
11	Dout5	30	Dout6
12	Dout7	31	Dout8
13	Dout9	32	Dout10
14	Dout11	33	Dout12
15	Dout13	34	Dout14
16	Dout15	35	Dout16
17	CLK 1	36	CLK2
18	GATE1	37	GATE2
19	OUT1		

四、模拟输入信号的连接方式

4.1 单端输入方式：

KPCI-811卡均可按图4.1连接成模拟电压单端输入方式，16路模拟输入信号连接到CH1～CH16端，其公共地连接到AGND端，注意：为了抑制噪声干扰，所有信号源接地端都应该一点接**AGND**。(见图4.1)

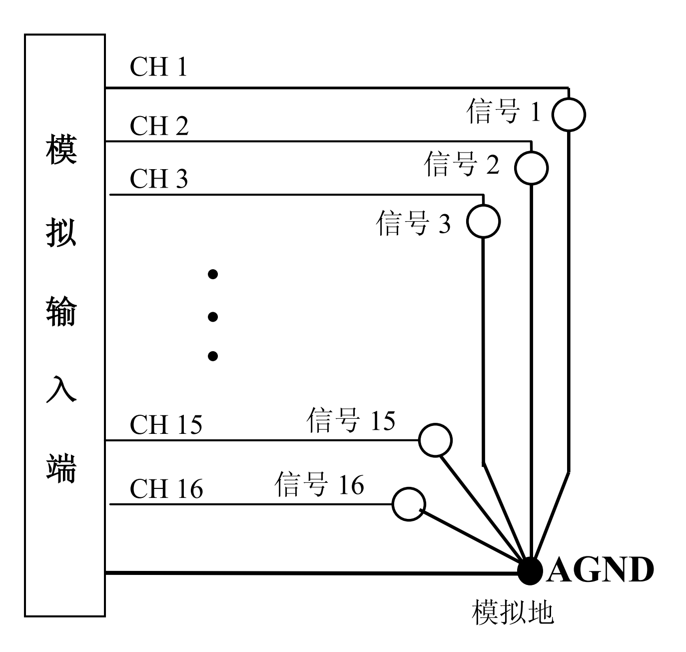


图 4.1 单端输入方式

4.2 双端输入方式：

KPCI-811卡可按图4.2连接成模拟电压双端输入方式，可以有效抑制共模干扰信号，提高采集精度。8 路模拟输入信号正端接到In1+～In8+端，其模拟输入信号负端接到In1-～In8-端， (见图4.2)

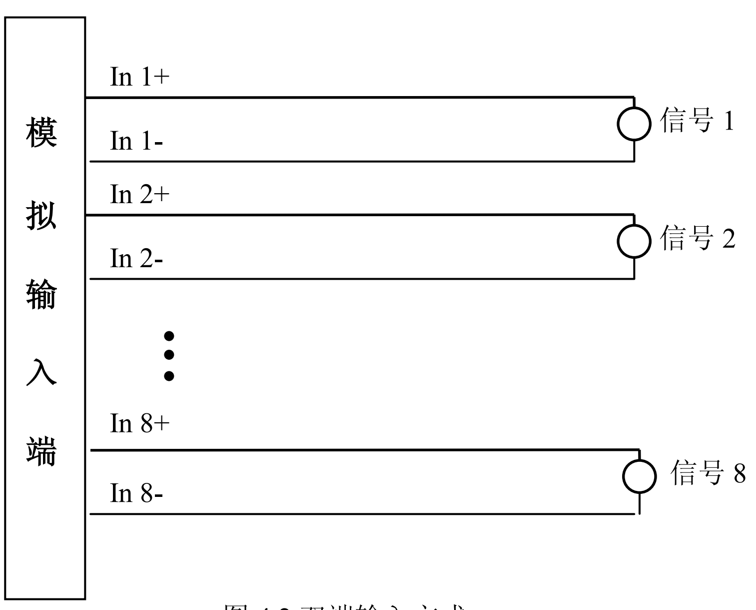


图 4.2 双端输入方式

五、模拟输出信号的连接方式

5.1 电压输出连接

KPCI-811 卡用于电压输出时，负载一端接电压输出端（DA_V_Out），另一端接模拟地，提供给负载的最大电流为 10mA，电压输出不需要外接电源。

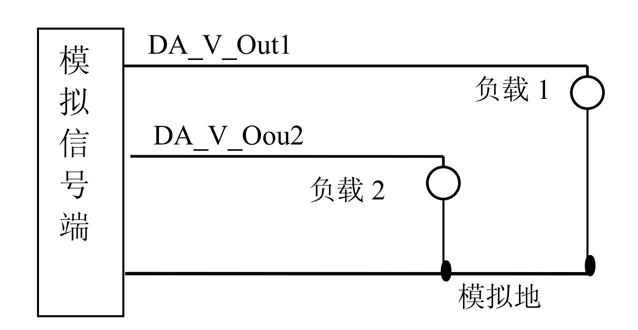


图 5.1 电压输出方式

5.2 电流输出连接

电流输出方式可以直接驱动 II 、III型执行仪表；供电电源可以使用本卡提供的＋15V。也可扩展外接机外＋24V 电源。其连接使用方法见下图

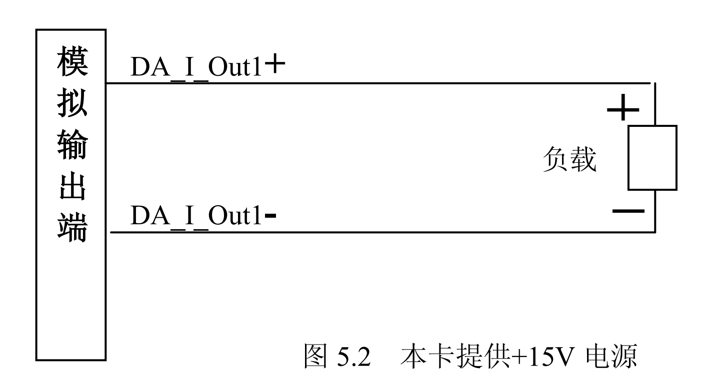


图 5.2 本卡提供+15V 电源

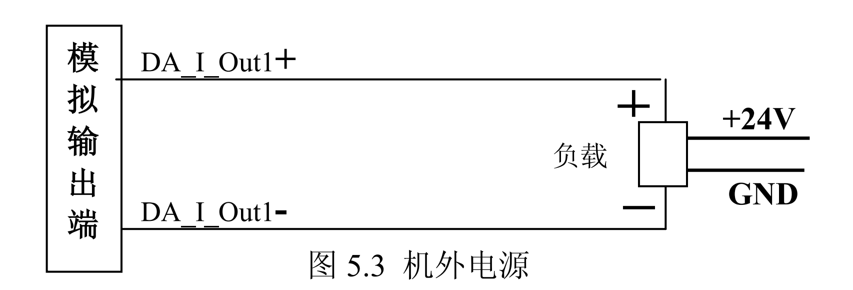


图 5.3 机外电源

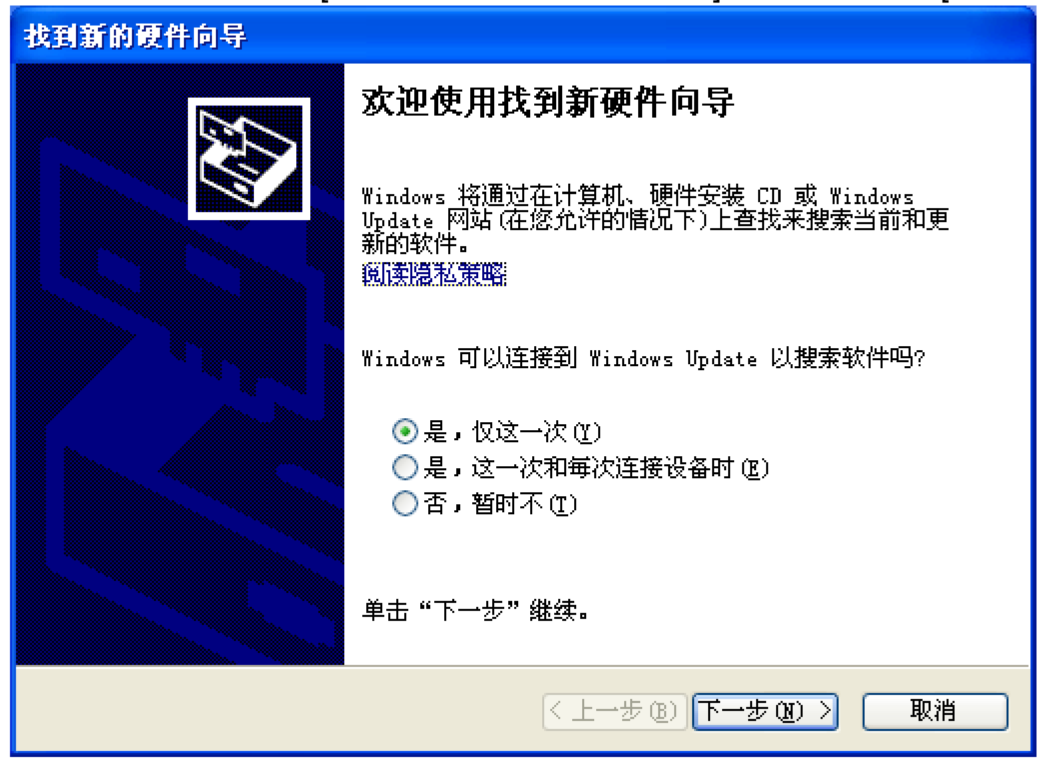
特别注意：电流输出时，不要将负载悬空。对于未使用的通道，请将跳线设置成电压输出方式。

第三章 KPCI-811 设备驱动程序安装（WindowsXP 环境下）

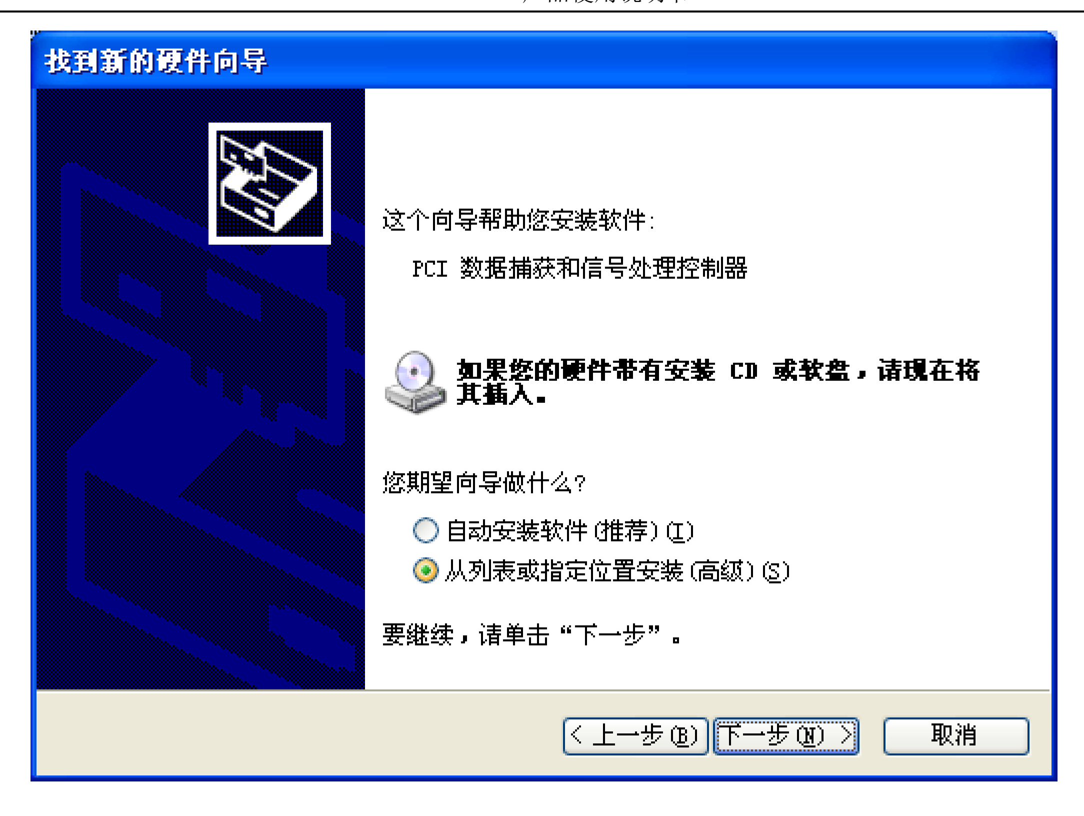
一、安装步骤

第一步 将 KPCI-811 卡按硬件要求插入计算机主板上的任意一个 PCI 插槽中，并将其固定好，连接好其外接设备后，打开计算机电源，启动 Windows2000(XP)系统。

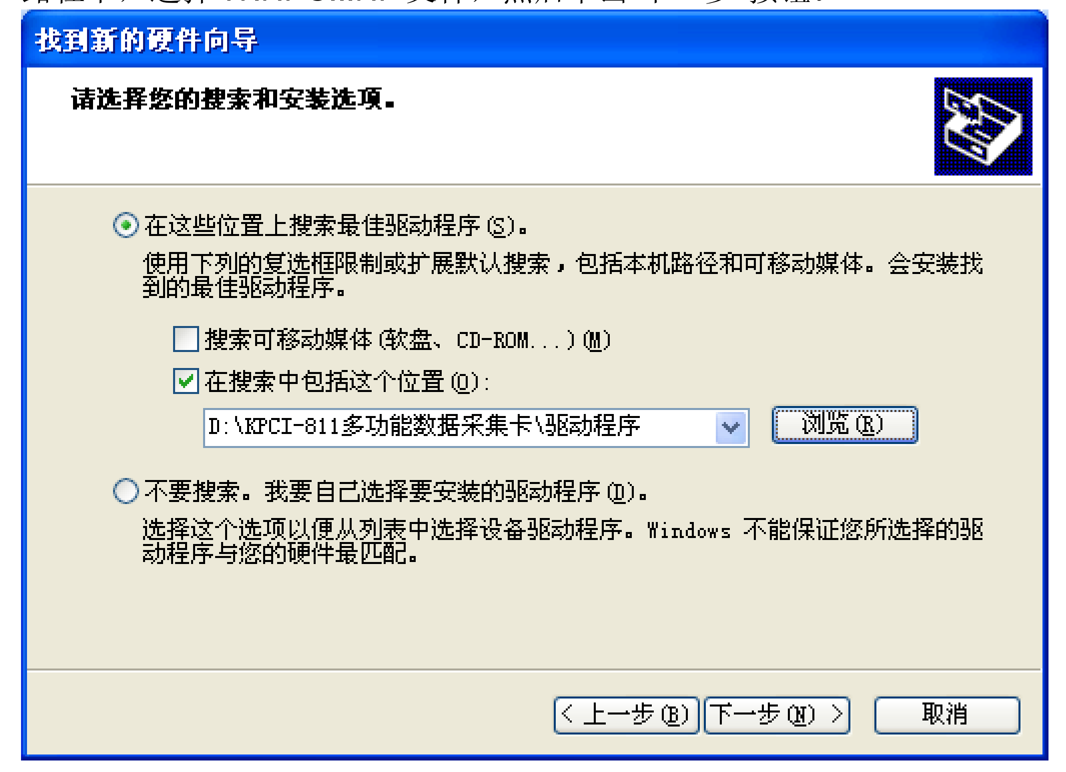
第二步 如果您正确地插好了 PCI 设备，Windows 系统在启动过程中便会发现这个新的 PCI 设备，并弹出 [欢迎使用找到新的硬件向导] 对话框，单击 [下一步] 按钮



第三步 弹出下面对话框，按照图示选择，然后单击 [下一步] 按钮。



第四步 将路径定位在光盘上的\PCI 总线测控板卡\KPCI-811 多功能数据采集卡\驱动程序路径下，选择 WINPCI.INF 文件，然后单击“下一步”按钮。



第五步 当出现下面的画面时，安装完成，单击"完成"按钮



第六步 KPCI-811 通用驱动和接口程序及测试、示范程序的安装：

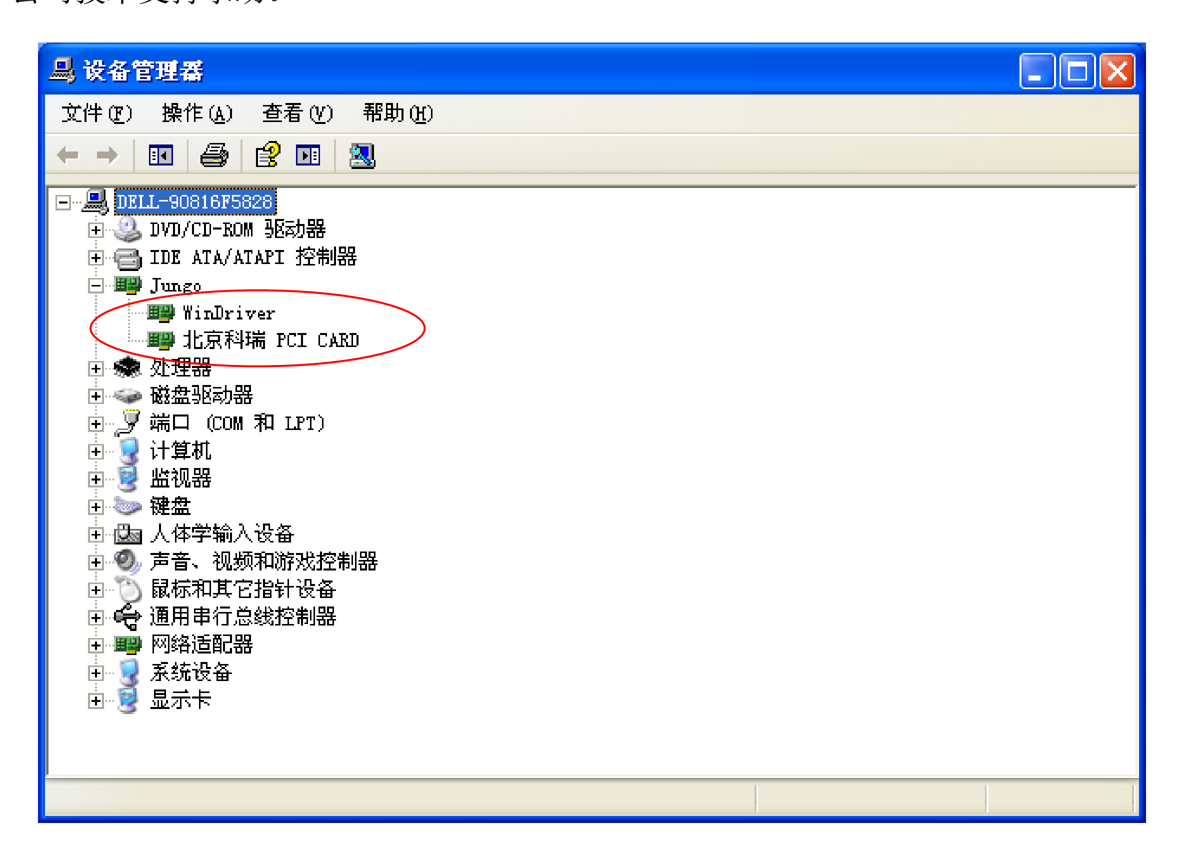
完成了上述驱动程序的安装后。接下来安装应用程序软件包。进入"我的电脑"，将路径定位在光盘上的\PCI 总线测控板卡\KPCI-811 多功能数据采集卡\驱动程序目录下，双击"PciSetup.exe"文件，把软件包安装在你指定的路径下的文件夹内，其中包括了通用驱动、动态连接库、测试和编程示例程序。根据需要解压缩 VB，VC 软件包，可以看到动态连接库、测试和编程示例程序。使用测试程序可以验证板卡的功能。VC，VB 的编程演示举例可以帮助你编写自己的应用程序。



至此，**KPCI-811** 卡的驱动程序安装成功。

二、安装结果验证

进入 WindowsXP[设备管理器] 窗口，在 [本地计算机] 列表中点击 [系统设备] 在展开的子列表中检查是否有“北京科瑞 PCI CARD 和 WinDriver”等显示，若有，表示 KPCI-811 卡的驱动程序已成功安装，否则，说明您的安装过程出现了问题，请试着再安装，或向我公司技术支持求助。



三、 PCI 设备软件测试系统的介绍

怎样进入测试系统：当您正确完成了上述的工作，您便可以在 Windows 的系统菜单中,寻找并进入您安装的路径。运行 KPCI-811 卡的测试程序。

KPCI-811_VC 程序用于测试 KPCI-811 卡在 VC 环境下的工作状态。

KPCI-811_VB 程序用于测试 KPCI-811 卡在 VB 环境下的工作状态。

第四章　函数模块调用说明及编程实例

光盘中提供了两个分别在 VB 和 VC 下开发的示例程序，给出了 KPCI-811 卡的各种触发方式、相关参数的设置过程和数据读取方法。VB 示例程序以流程图形式说明了本卡 AD 部分的工作过程和顺序，包括卡的打开与关闭。对使用 VC 的用户也可以利用此 VB 示例程序了解卡的功能和使用方法。用户可以参照相应程序段根据实际需要利用函数库中提供的函数设计自已的软件，而对数据采集过程中不处理其他任务的用户，也可以直接利用示例程序完成数据。初次使用动态连接库的用户还可以在程序中找到动态连接库的调用方法。为方便用户分析，示例程序以工程的形式提供了所有的资源和代码。用户在编译时需要注意重新指定动态连接库的路径，或把函数库拷到指定的位置。

一 A/D 采集过程流程图

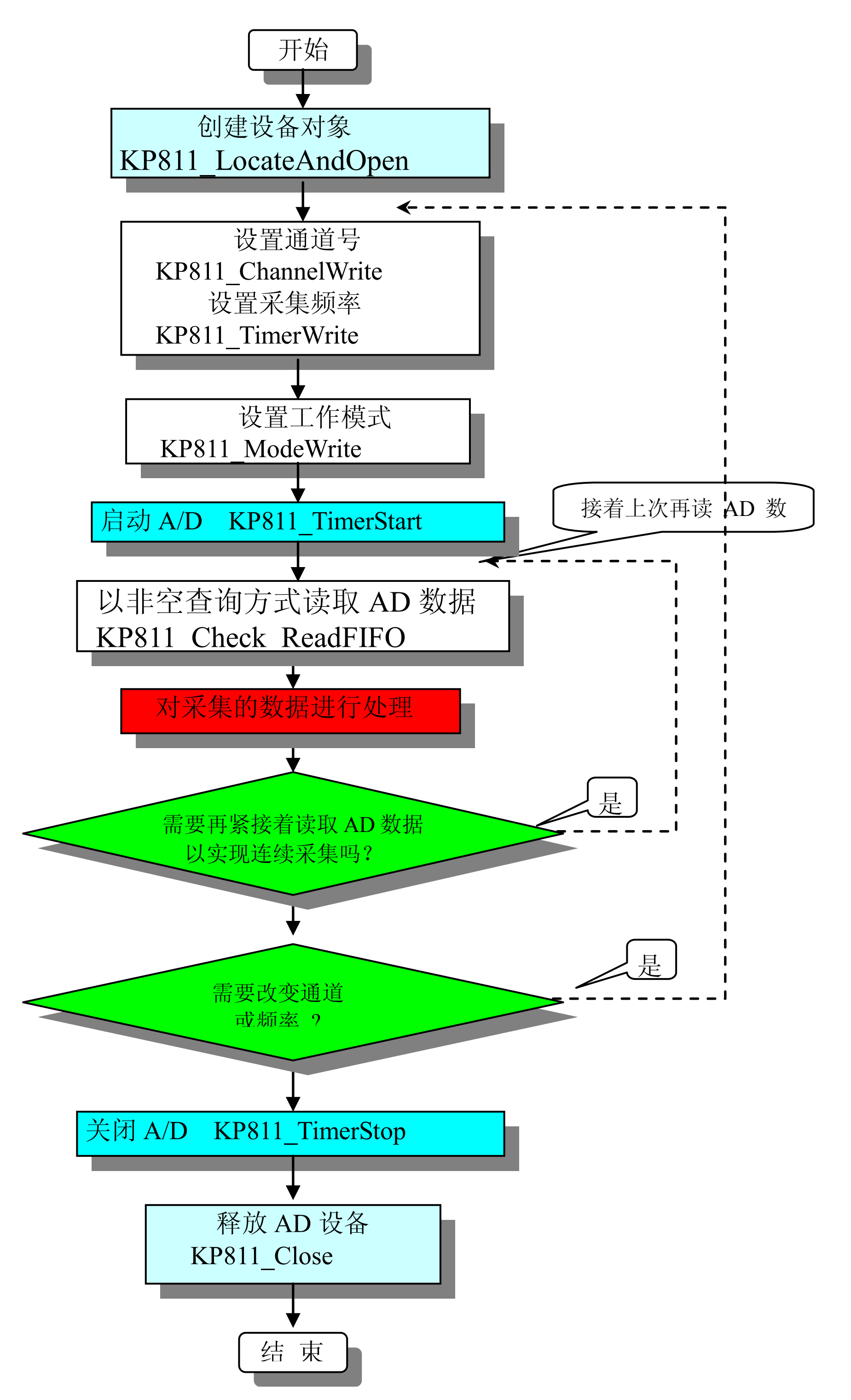


图 3.1 非空查询方式 AD 采集过程

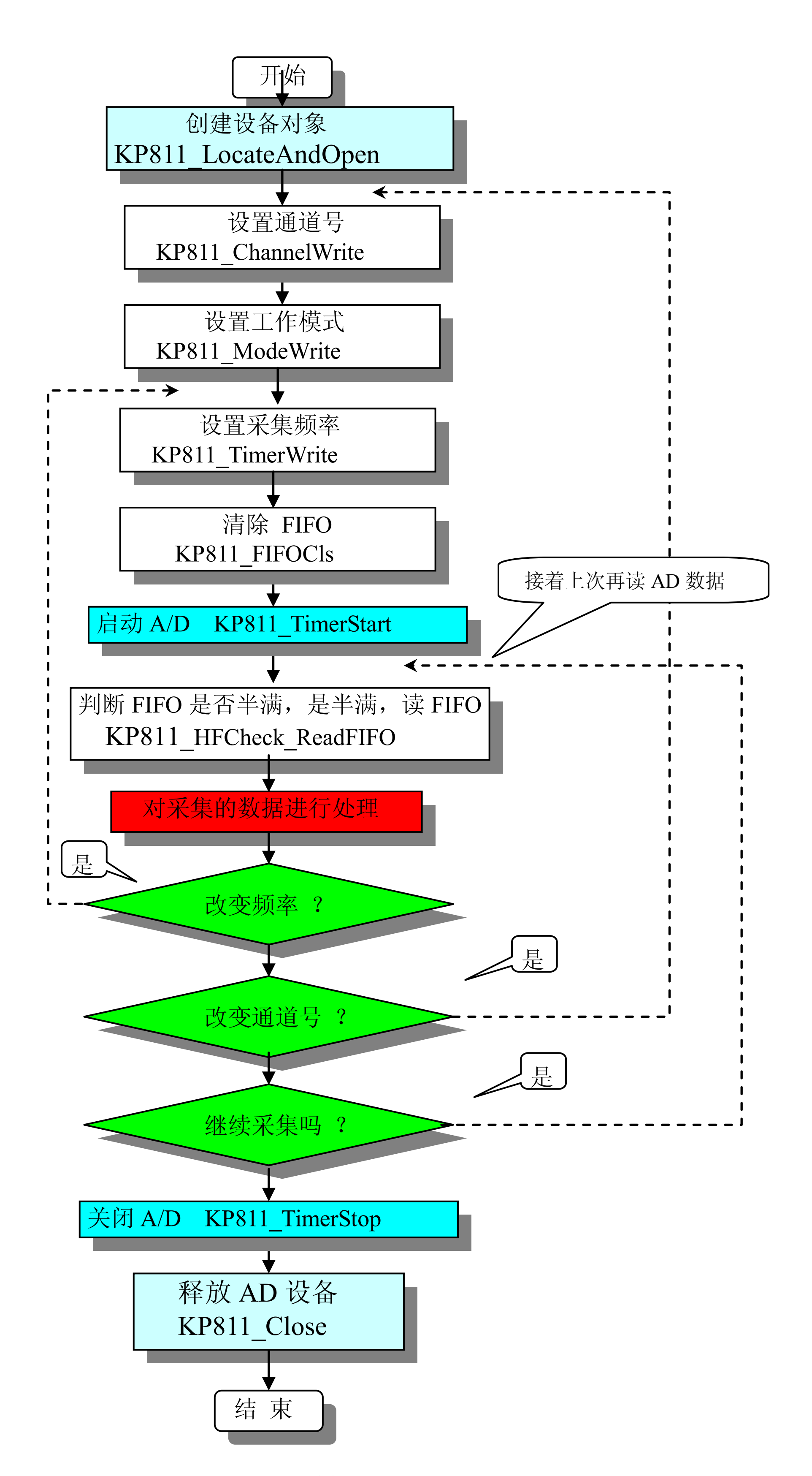


图 3.2 半满方式 AD 采集实现过程

二、 函数说明：

KPCI-811 函数简表	
函数名	函数功能
KP811_LocateAndOpen	创建 PCI 设备对象
KP811_TimerWrite	设置采集频率
KP811_ModeWrite	设置工作模式
KP811_ChannelWrite	设置 A/D 通道号
KP811_CheckSF_ReadFIFO	软件触发 A/D 并读数据
KP811_SoftConvert	软件触发 A/D 结果存入 FIFO
KP811_TimerStart	定时方式下启动 A/D
KP811_TimerStop	定时方式下暂停 A/D
KP811_FIFOCls	刷新 FIFO
KP811_HFCheck_ReadFIFO	读半满 FIFO
KP811_HFSFCheck_ReadFIFO	软件触发 A/D 直到半满
KP811_ReadFIFO	读FIFO中的数据
KP811_Check_ReadFIFO	检查 FIFO 非空标志的读
KP811_ReadStatus	读 FIFO 状态
KP811_Input	开关量输入
KP811_Output	开关量输出
KP811_8254_CTRL_Write	设置 8254 定时计数工作模式
KP811_8254_Value_Read	读定时/计数器值
KP811_8254_Value_Write	设置 8254 时间常数
KP811_DAC	模拟量输出
KP811_Close	释放设备对象所占用的系统资源

1、创建设备对象

Visual C++ & C++Builder：

IO_HANDLEL KP811_LocateAndOpen (DWORD card_no)

Visual Basic：

Declare Function KP811_LocateAndOpen Lib "IODLL.dll" (ByVal card_no As Long) As Long

功能：该函数负责创建 PCI 设备对象，并返回其设备对象句柄。

参数：card_no：指定 KPCI-811 卡在计算机的序号，从 0 开始的整数

IO_HANDLE 设备 ID(Identifier)标识号。当向同一个 Windows 系统中加入若干相同类型的 PCI 设备时，驱动程序将以该设备的 card_no 标识值为名称后缀的标识符来确认该设备。并且是计算机系统中的唯一标识；对本卡操作的所有函数都要使用它。

返回值：如果执行成功，则返回设备对象(句柄) 标识值；
如果未成功， 则返回 NULL。

2、设置 AD 采集频率

Visual C++ & C++Builder：

void KP811_TimerWrite(IO_HANDLE hIO, BYTE m_TIMER*)*

Visual Basic：

Declare Sub KP811_TimerWrite Lib "IODLL.dll" (ByVal hIO As Long
, ByVal TIMER As Integer)

功能：定时触发方式下时间常数写入函数，该函数用于在定时触发方式下确定 AD 转换器的采样周期。

参数：hIO 为 KP811_LocateAndOpen 函数调用时返回的句柄值。

m_TIMER 为定时时间常数。

m_TIMER 的最大取值为 1023。最小值与分频系数有关，总的原则是采用周期应不小于 1 微秒。AD 采集周期 T 除和定时时间常数 m_TIMER 有关之外，还与分频系数 K 有关（分频系数参见 KP811_ModeWrite 函数）。

周期 T=（m_TIMER+1）*K*16*2/33 单位为微秒。

3、设置工作模式

Visual C++ & C++Builder：

Void KP811_ModeWrite(IO_HANDLE hIO,BYTE m_ENABLE
,BYTE m_SINGLE
,BYTE m_SCAN
,BYTE m_TIMES
,INT m_FREQ)

Visual Basic：

Declare Sub KP811_ModeWrite Lib "IODLL.dll" (ByVal hIO As Long
,ByVal m_ENABLE As Byte
,ByVal m_SINGLE As Byte
,ByVal m_SCAN As Byte
,ByVal m_TIMES As Byte
, ByVal m_FREQ As Integer)

功能：设置多功能工作方式

参数：hIO： 为 KP811_LocateAndOpen函数调用时返回的句柄值。

m_ENABLE: 表示触发方式选择参数：0：软件触发
1：外部触发
2：定时触发。

m_SINGLE：单端/双端输入选择： 0：单端输入
1：双端输入

m_SCAN： 0：表示禁止通道扫描
1：允许通道扫描

m_TIMES：放大倍数 可设置值为：1，2，4，8

m_FREQ：定时器分频系数设置参数，可设置值为：2，4，8，16

注意事项：

1、 在定时触发方式下，AD 根据指定频率采集数据。如不能及时从 FIFO 中读取数据，FIFO 将会由于填满，从而阻止写入新的数据，会造成数据丢失的现象。可以在设定采样的点数（或采样时间结束）时；暂时把 A/D 工作模式设置为软件触发并禁止通道扫描。

2、 无论本卡 AD 部分工作在哪一种触发方式下，都可以设置为自动通道扫描方式。使用时应注意数据的正确分配，尤其要小心 FIFO 已满丢失数据，造成数据顺序的混乱。

3. 在定时触发和外部触发方式下使用通道扫描采集结束后，最后的通道号是随机的，也不要考虑 FIFO 中还有多少个数未取。强烈建议在启动新一轮数据采集前按下列顺序重新设置相应参数，步骤①不要省略。

① **工作模式字清 0**　　　KP811_ModeWrite(IO_HANDLE hIO,0,0,1,2)

② 送新的（最高）通道号；*KP811_ChannelWrite(IO_HANDLE hIO,BYTE m_RSW)*

③ 设定采样频率；*KP811_TimerWrite(IO_HANDLE hIO, BYTE m_TIMER)*

④ 清除 FIFO 中剩余的数据；*KP811_FIFOCls(IO_HANDLE hIO)*

⑤ 送实际的工作模式字; KP811_ModeWrite(IO_HANDLE hIO
,BYTE m_ENABLE
,BYTE m_SINGLE
,BYTE m_SCAN
,BYTE m_TIMES
,INT m_FREQ)

⑥ 启动 AD 转换；　*KP811_TimerStart (IO_HANDLE hIO)*

⑦ 读取采集结果。*KP811_HFCheck_ReadFIFO(IO_HANDLE hIO
,WORD LengthFIFO
,WORD *P)*

4. m_SINGLE 的设定值应与卡上的 JP1 跳线一致。(单端/双端输入)
5. AD 卡的 A/D 芯片转换数据输出有 3 个转换周期的延迟，FIFO 中的前三个数据应舍弃。

4、设置 A/D 通道号

Visual C++ & C++Builder：

void KP811_ChannelWrite(IO_HANDLE hIO,BYTE m_RSW)

Visual Basic：

***Declare Sub KP811_ChannelWrite Lib "IODLL.dll" (ByVal hIO As Long
,ByVal channel As Byte)***

功能：设置通道号

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。
m_RSW：为非扫描方式下的模拟量输入通道号。
在扫描方式下，m_RSW 为扫描的最高通道号。

注意事项：

1. 在非扫描方式下，通道号写入后一直保持不变，直到重新使用该函数写入新的值。
2. 在扫描方式下，由该函数指定扫描的最高通道号。该方式下 A/D 每采集一次结束后，自动切换到下一通道，扫描顺序采用递减顺序。
3. 单端方式下 m_RSW 的取值范围为 0~15 之间的整数。
双端方式下 m_RSW 的取值范围为 0~7 之间的整数。

5、软件触发 A/D 并读数据

Visual C++ & C++Builder：

WORD KP811_CheckSF_ReadFIFO(IO_HANDLE hIO)

Visual Basic：

Declare Function KP811_CheckSF_ReadFIFO Lib "IODLL.dll" (ByVal hIO As Long) As long

功能：软件触发启动A/D一次，并读数据

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

该函数用于在软件触发下启动一次新的 AD 转换并返回一个双字节。该函数两次调用之间的最短时间为 500ns。

6、软件触发 A/D 直到半满

Visual C++ & C++Builder:

WORD KP811_HFSFCheck_ReadFIFO (IO_HANDLE hIO,
WORD FIFOlength,
WORD *point)

Visual Basic:

Declare Function KP811_HFSFCheck_ReadFIFO Lib " IODLL.dll " (
ByVal hIO As Long,
ByVal FIFOlength As Long,
ByRef point As Long) As Long

功能：软件触发启动A/D一直达到半满，并读数据

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

FIFOlength：表示 FIFO 的大小；取值为（1,2,4,8,16…）分别表示 1K,2K,4K,8K,16K

Point：数据缓冲区首指针，调用该函数之前应申请大于 FIFO 的容量的存储空间 如 FIFO 为 4K，则应申请大于 2K 的内存空间。
该函数首先检查 FIFO 半满标志，如未半满，则再一次启动 A/D。直到 FIFO 中数据达到半满。然后将数据一次性读入 Point 指向的缓冲区。

7、软件触发 A/D 结果存入 FIFO

Visual C++ & C++Builder:

void KP811_SoftConvert(IO_HANDLE hIO)

Visual Basic:

Declare Sub KP811_SoftConvert Lib "IODLL.dll" (ByVal hIO As Long)

功能：软件触发启动A/D一次，并将结果送入FIFO

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

注意事项: 此函数只是启动 A/D 一次，并将结果存如 FIFO，但并未读回内存。

8、定时方式下启动 A/D

Visual C++ & C++Builder:

void KP811_TimerStart (IO_HANDLE hIO)

Visual Basic:

Declare Sub KP811_TimerStart Lib "IODLL.dll" (ByVal hIO As Long)

功能：定时方式启动A/D

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

9、定时方式下暂停 A/D

Visual C++ & C++Builder:

void KP811_TimerStop (IO_HANDLE hIO)

Visual Basic:

Declare Sub KP811_TimerStop Lib "IODLL.dll" (ByVal hIO As Long)

功能：定时方式暂停A/D

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

10、刷新 FIFO

Visual C++ & C++Builder：

void KP811_FIFOCls(IO_HANDLE hIO)

Visual Basic：

Declare Sub KP811_FIFOCls Lib " IODLL.dll " (ByVal hIO As Long)

功能：清除FIFO

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

利用该函数可随时清除 FIFO 中未取走的转换结果，确保读到的数据是最新的转换结果。该函数一般在定时或外部触发方式下启动"新一轮"转换之前调用。

11、读半满 FIFO

Visual C++ & C++Builder：

```
DWORD   KP811_HFCheck_ReadFIFO(IO_HANDLE hIO，
                               WORD FIFOregion,
                               WORD *P)
```

Visual Basic：

```
Declare Function KP811_HFCheck _ReadFIFO Lib "IODLL.dll"(
                                  ByVal hIO As Long
                                  ,ByVal FIFOregion As Long
                                  ,ByRef P As Long) As Long
```

功能：读一半FIFO缓冲区

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

FIFOregion：表示 FIFO 的大小；取值为（1,2,4,8,16…）分别表示 1K,2K,4K,8K,16K

P：数据缓冲区首指针，调用该函数之前应申请大于 FIFO 的容量的存储空间如 FIFO 为 2K，则应申请大于 2K 的内存空间。

该函数首先检查 FIFO 半满标志，如未半满，则等待。直到 FIFO 中数据达到半满

注意事项：

当 FIFO 为空且 AD 无触发条件时，则该函数会一直等待下去。直到 FIFO 中数据达到半满。用户使用该函数时应负责正确的操作时序和 FIFOregion 的设置值。

12、读FIFO中的数据

Visual C++ & C++Builder：

WORD KP811_ReadFIFO(IO_HANDLE hIO)

Visual Basic：

Declare Function KP811_ReadFIFO Lib "IODLL.dll"(ByVal hIO As Long) As Integer

功能：读FIFO中的数据。

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

注意事项：

该函数 ***KP811_Check_ReadFIFO*** 函数的差别是读 FIFO 前不检查 FIFO 的空标志，所以用户使用该函数时应确保 FIFO 中不为空。

13、读FIFO中的数据

Visual C++ & C++Builder：

WORD KP811_Check_ReadFIFO(IO_HANDLE hIO)

Visual Basic：

Declare Function KP811_Check_ReadFIFO Lib "IODLL.dll"(ByVal hIO As Long) As Integer

功能：带检查FIFO空标志的读FIFO函数

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

注意事项：

该函数首先检查 FIFO 是否为空，若 FIFO 为空则等待，直至 FIFO 中存入数据后，再从 FIFO 中读一个数（16 位）。当 FIFO 为空且 AD 无触发条件，则该函数会一直等待下去。用户使用该函数时应选择好正确的操作时序。

14、读FIFO状态

Visual C++ & C++Builder：

Long　KP811_ReadStatus(IO_HANDLE hIO)

Visual Basic：

Declare Function KP811_ReadStatus Lib "IODLL.dll"(ByVal hIO As Long) As Long

功能：读FIFO状态

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

该函数读取 FIFO 的状态，返回 FIFO 的三个状态标志。

	0	1
D0	FIFO 空	FIFO 非空
D1	FIFO 已半满	FIFO 未半满
D2	FIFO 已全满	FIFO 未全满

1、 返回值的 D0 位为空标志，当 FIFO 为空时该位为 0，FIFO 非空时该位为 1。
2、 返回值的 D1 位为半满标志，当 FIFO 中数达到容量的一半以上（包括一半）时该位为 0，否则该位为 1。
3、 返回值的 D2 位为全满标志，当 FIFO 中数达到满容量时该位为 0，否则该位为 1。
4、 当全满标志为 0 时，FIFO 自动禁止写入新的转换结果，所以此时可能丢掉新的采样值。所以用户应及时读取 FIFO 的数，防止 FIFO 全满。

15、模拟量输出

Visual C++ & C++Builder：

Void KP811_DAC(IO_HANDLE hIO,short channel, long val)

Visual Basic：

Declare Sub KP811_DAC Lib "IODLL.dll" (ByVal hIO As Long
,ByVal channel As Interger
,ByVal Val As Long)

功能：模拟量输出

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

channel：指定 D/A 的通道号，取值为 0 或 1 分别对应 D/A 的 0 和 1 通道

Val：12 位(二进制)的数值，取值范围应在 0~4095 之间。

16、开关量输入

Visual C++ & C++Builder:

WORD KP811_Input(IO_HANDLE hIO)

Visual Basic:

Declare Function KP811_Input Lib "IODLL.dll" (ByVal hIO As Long) As Long

功能：负责将 PCI 设备上的 16 路开关量输入状态读入内存。

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

该函数返回指定 AD 卡的当前开关量输入口上状态。低 16 位有效。分别对应于 Din1-Din16。

17、开关量输出

Visual C++ & C++Builder:

Void KP811_Output(IO_HANDLE hIO,short channel,bool states,bool initop,long val)

Visual Basic:

Declare Sub KP811_Output Lib "iodll.dll" (ByVal hIO As Long
,ByVal channel As Integer
, ByVal states As Long
, ByVal initop As Long
, ByVal val As Long)

功能：16 路开关量输出。

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

Initop：　当为逻辑值“true”时，指定卡该函数为按“字节” 方式输出
当 initop 为逻辑值“false”时，指定卡该函数为按“通道” 方式输出

channel：通道输出方式下的通道号，通道号取值范围为(0-15)

states：　通道输出方式下，指定通道的目标状态。
当逻辑值“true”时，相当于对指定的通道写“1”。
否则，写“0”

val：　“字节”输出方式下，指定 16 个通道的目标状态。
该参数取值范围应在 0~~65535 之间，即低 16 为有效。

注意事项:

1. 当首次调用 KP811_Outpout 时，参数 initop 必须为“true”。
2. 在通道输出方式下，只对指定的通道号输出操作，而不影响其他通道。通道号由参数 channel 指定，状态由参数 states 指定。

18、设置 8254 定时计数工作模式

Visual C++ & C++Builder:

Void KP811_8254_CTRL_Write(IO_HANDLE hIO,BYTE timer,BYTE modeword)

Visual Basic:

Declare Sub KP811_8254_CTRL_Write Lib "IODLL.dll" (ByVal hIO As Long
,ByVal timer As Byte
, ByVal modeword As Byte)

功能：设置 8254 定时/计数工作模式

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。

timer：指定 8254 的通道号，取值为 0 或 1
modword：设定 8254 的工作模式；取值为 0,1,2,3,4,5;分别对应 8254 的模式 0，模式 1，模式 2，模式 3，模式 4，模式 5，

注意事项：

所有计数器都为 2 进制方式。时间常数写入、读出均为 16 位。

19、设置 8254 时间常数

Visual C++ & C++Builder：

Void KP811_8254_Value_Write(IO_HANDLE hIO,BYTE timer ,unsigned short timeinterval)

Visual Basic：

Declare Sub KP811_8254_Value_Write Lib "IODLL.dll" (ByVal hIO As Long ,ByVal timer As Byte , ByVal timeinterval As Interger)

功能：设置 8254 定时/计数器时间常数

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。
timer：指定 8254 的通道号，取值为 0 或 1
timeintertval：8254 定时/计数器的时间常数；16 位二进制无符号数 取值范围为：0~65535

20、读定时/计数器值

Visual C++ & C++Builder：

Word KP811_8254_Value_Read(IO_HANDLE hIO,BYTE timer)

Visual Basic：

Declare Function KP811_8254_Value_Read Lib "IODLL.dll" (ByVal hIO As Long ,ByVal timer As Byte) As Long

功能：读 8254 计数/定时器的数值

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。
timer：指定 8254 的通道号，取值为 0 或 1

注意事项：

返回值为 16 位二进制无符号整数，读出值的有效范围为：0~65535。

21、释放设备对象所占的系统资源及设备对象

Visual C++ & C++Builder：

void KP811_Close(IO_HANDLE hIO)

Visual Basic：

Declare Sub KP811_Close Lib "IODLL.dll" (ByVal hIO As Long)

功能：释放指定的A/D设备

参数：hIO：为 KP811_LocateAndOpen函数调用时返回的句柄值。
程序结束时应调用该函数。

第五章　KPCI-811 多功能卡的校准、保修和注意事项

一　应用注意事项

在公司售出的产品包装中，用户将会在资料光盘中找到本卡的说明书。

在使用KPCI-811板时，应注意以下问题：

① KPCI-811卡正面的IC芯片不要用手去摸，防止芯片受到静电的损害。

② 在使用KPCI-811卡时，可通过K-801E等信号调理端子板与现场信号连接。

③ 用户务必注意电源的开关顺序，使用时要求先开主机电源，后开信号源的电源；先关信号源的电源，后关主机电源。

二、校准

KPCI-811板出厂时已经校准，只有当用户使用一段时间后，或者用户认为需要时才做校准。下面以0～10V量程为例，说明校准过程：

准备一块4位半精度以上数字电压表，安装好KPCI-811，打开主机电源，预热10分种。

1、A/D的校准(以单极性输入为例)：

1） ① 零点调整

放大器增益选择1倍。选模拟输入的任意两个通道，比如CH1、CH2通道，将CH0通道输入接0伏电压(AGND),在WINDOWS下运行KPCI-811 测试程序,选择1通道，单通道数据采集方式，采集开始后，调整电位器W7，使显示值为 0。

② 满度调整

将 CH1接正满度电压+9998 mV，在选择2通道，采集程序开始采集后，调整电位器W9，使显示值为 4095。

重复以上步骤，直到满足要求为止。

2） 如果放大器增益选择2，4，8倍，在相应的通道上，也要接入相应的满度电压。（5 ,2.5, 1.25 或 0.625V）

2、 D/A的校准（以电压输出0~10V为例）：

① 零点调整：将数字万用表的红表笔接DA_V_Out1，黑表笔接模拟地。运行测试程序，在DA1的数据输入框中输入“0”，按“通道0发送”，调整电位器W4，使数字万用表显示为“0”。

② 满度调整：在DA1的数据输入框中输入“4095”，按“通道0发送”，调整电位器W3，使数字万用表显示为“1 0V”。

重复以上步骤，直到满足要求为止。

三、保修

KPCI-811卡自出厂之日起，两年内凡用户遵守贮存，运输和使用要求，而产品质量低于技术指标的，凭保修卡免费维修。因违反操作规定和要求而造成损坏的，需交纳器件维修费。

四、产品配套清单：

4.1 KPCI-811多功能数据采集卡壹块。
4.2 北京科瑞兴业公司产品光盘壹张。
4.3 37芯D型插头壹套。
4.4 40芯扁平带缆插头壹套。